

REMARKS

Claims 1-13 are pending in this application. Further reconsideration is requested based on the following remarks.

Response to Arguments:

The Applicants appreciate the consideration given to their arguments, and particularly the withdrawal of the rejection of claim 9 under 35 U.S.C. § 103 as being unpatentable over US Patent No. 5,298,006 to Miyajima in view of US Patent No. 5,319,778 to Catino. The Applicants also acknowledge with appreciation the effort put into the final Office Action to illustrate the reasoning behind the other rejections. The Applicants, however, are disappointed that their arguments were not found to be persuasive with respect to the other rejections, and hope for favorable reconsideration.

In several embodiments, the claimed invention provides a numerical controller with the capability of efficiently executing a machining program including a branch instruction, as described in the specification at page 1, lines 5 and 6. The final Office Action notes an excerpt of the specification at page 8 that describes, in one embodiment, how this comes to be. In particular, as described in the specification at page 7, lines 9, 10, and 11:

As shown in FIGS. 4 and 5, it is not necessary to input/output data of the machining program sequentially from the beginning to the end, contrary to the conventional art.

This excerpt of the present specification, focused upon so adroitly by the final Office Action, points up one aspect in which the claimed invention differs from the cited references like, for example, Park. The process control algorithm of Park, in particular, in contrast to the claimed invention, cannot be divided, let alone run in any order other than serial order. In particular, as described at column 5, lines 8-12 of Park:

To convert the first block diagram list to an executable form, the identification numbers should be reallocated in the order of actual execution. That is, the block diagram list should be rearranged in the order of execution.

Since, in Park, the block diagram list should be rearranged in the order of execution before it is run, Park is not "dividing the machining program so that divided portions of the machining program are stored in respective ones of the input/output units," as recited in, for example, claim 9.

Park, furthermore, renumbers all of the function blocks so that the identification numbers may be *continuous* without any *vacancy*. In particular, as described at column 5, lines 25-28 of

Park:

At step 21, all function blocks in the function block diagram are renumbered so that the identification numbers may be continuous without any vacancy. The identification numbers after the renumbering at step 21 are shown in FIG. 7c.

Since, in Park, the function blocks in the function block diagram are renumbered so that the identification numbers may be continuous without any vacancy, Park is not "dividing the machining program so that divided portions of the machining program are stored in respective ones of the input/output units," as recited in, for example, claim 9.

The final Office Action asserts at page 10, lines 8 and 9 that:

Figs. 9a and 9b clearly shows divided portions of the machining program, for example, "ADD_I" and "SQRT".

This is submitted to be incorrect. Figs. 9a and 9b of Park, rather, show block diagram lists of a process control algorithm and structured control codes, respectively. In particular, as described in Park at column 2, lines 30-36:

FIG. 9a shows a block diagram list of a process control algorithm represented as a function block diagram as shown in FIG. 7a;
FIG. 9b shows a block diagram list of structured control codes which correspond to the directed graph shown in FIG. 7d.

Figs. 9a and 9b, furthermore, represent the input and outputs of a conversion process that rearranges an original block diagram list (a first list) corresponding to a function block diagram edited according to the order of execution into a block diagram list (a second list) corresponding to structured control codes, not a machining program, divided or otherwise, at all. In particular, as described at column 2, lines 30-36 of Park:

Upon selecting the "code generation" menu included in the main menu shown in FIG. 5, the control algorithm in the form of a function block diagram is converted to structured control codes at step 20. The actual conversion process includes arranging an original block diagram list (a first list) corresponding to a function block diagram edited according to the order of execution to thereby provide a block diagram list (a second list) corresponding to structured control codes.

Since, in Park, the original block diagram list (a first list) corresponding to a function block diagram edited according to the order of execution is rearranged into a block diagram list (a second list) corresponding to structured control codes, Park is not "dividing the machining program so that divided portions of the machining program are stored in respective ones of the input/output units," as recited in, for example, claim 9.

Furthermore, in Park, function block records 101-108 and 111-118, shown in Figs. 9a and

9b of Park are minimum elements that are not divisible. The function block records 101-108 in the first block diagram list shown in Fig. 9a are rearranged to become the function block records 111-118 in the second block diagram list shown in Fig. 9b by the procedure shown in Figs. 10a-10c.

In the claimed invention, in contrast, each of the input/output units include program block data which are obtained by dividing a machining program and which are thus capable of being divided or combined further, if necessary.

The final Office Action asserts further at page 10, lines 9-13 that:

Figs. 9a and 9b clearly shows a "machining program is divided into a plurality of program blocks and additional information is added to each of the program blocks to be associated therewith to form input/output units having the same number as that of the program blocks in one-to-one relation with the program blocks."

This is submitted to be incorrect. Figs. 9a and 9b of Park, rather, show a function block diagram edited according to the order of execution being rearranged into a block diagram list corresponding to structured control codes, not a machining program, divided or otherwise, at all. No division is shown in Figs. 9a and 9b of Park at all, contrary to the assertion in the final Office Action. Figs. 9a and 9b of Park, rather, show *combination*, not division.

The final Office Action asserts further at page 10, lines 16-19 that:

The examiner respectfully submits that elements 101, 102, etc. shown above with program blocks "ADD_I", "SQRT", etc, clearly divided among elements 101, 102, etc, is each of the input/output units storing program data obtained by dividing the machining program are stored in respective ones of the input/output units.

This is submitted to be incorrect. In Park, rather, each block record, i.e. "ADD_I", "SQRT", etc, is deleted from the first list and appended to the last block of the second list one by one until no block record is left in the first list or the stack is null. None of the block records are divided, in Park. The figures to which the final Office Action refers actually show a *combination* process, not "dividing the machining program so that divided portions of the machining program are stored in respective ones of the input/output units," as recited in, for example, claim 9. In particular, as described at column 6, lines 3-12 of Park:

As shown in FIG. 7a, the block ADD_I can be executed after VAR#1 and VAR#2. Therefore, it is deleted from the first list and appended as the next block of VAR#2 to the second list as shown in FIG. 10b. In this manner, each block record is deleted from the first list and appended to the last block of the second list one by one until no block record is left in the first list or the stack is null. After all the blocks are appended to the second list, the order of blocks in the second list (shown in FIG. 9b) corresponds to the identification number shown in 7d.

Since Park goes out of his way to ensure that each block record is deleted from the first list and appended to the last block of the second list one by one until no block record is left in the first list or the stack is null, Park is not "dividing the machining program so that divided portions of the machining program are stored in respective ones of the input/output units," as recited in, for example, claim 9. Further reconsideration is thus requested.

Claim Rejections - 35 U.S.C. § 102:

Claims 9, 11, and 12 were rejected under 35 U.S.C. § 102(b) as anticipated by Park et al., US 6,173,208 (hereinafter "Park"). The rejection is traversed.

Claim 9, in particular, recites:

A storage device storing a plurality of input/output units, each of the input/output units storing program data obtained by dividing the machining program so that divided portions of the machining program are stored in respective ones of the input/output units.

Park neither teaches, discloses, nor suggests a storage device storing a plurality of input/output units, each of the input/output units storing program data obtained by *dividing* the machining program so that divided portions of the machining program are stored in respective ones of the input/output units, as recited in claim 9. To the contrary, as Park describes at column 3, lines 27-33:

Referring to FIG. 2, at step 17, upon request from the system computer 1, input/output data stored in the hard disk 4 is transferred to the system computer 1 in the form of a database, to be used by the function block diagram editor. At step 18, the input/output data transferred to the system computer 1 is converted to a list of input/output function blocks.

Since, in Park, the input/output data transferred to the system computer 1 is converted to a *list* of input/output function blocks, *divided* portions of the machining program are not stored in respective ones of the input/output units.

Furthermore, as described in Park at column 4, lines 48-65:

Referring back to FIG. 2, remaining steps for producing structured control codes will be explained. Upon selecting the "code generation" menu included in the main menu shown in FIG. 5, the control algorithm in the form of a function block diagram is converted to structured control codes at step 20. The actual conversion process includes arranging an original block diagram list (a first list) corresponding to a function block diagram edited according to the order of execution to thereby provide a block diagram list (a second list) corresponding to structured control codes. In FIGS. 9a and 9b, an example of the block diagram list is illustrated as a data structure consisting of a plurality of function block

records 101-108, and 111-118 each of which includes a pointer to the previous function block record 101a, a pointer to the next function block record 101d, the name of the function block 101b, and information about the function block 101c, e.g., the numbers of inputs and outputs, and types of each input and output.

Since, in Park, the actual conversion process includes arranging an original block diagram list (a first list) corresponding to a function block diagram edited according to the order of execution to thereby provide a block diagram list (a second list) corresponding to structured control codes, *divided* portions of the machining program are not stored in respective ones of the input/output units, as recited in claim 9. Claim 9 is thus submitted to be allowable. Withdrawal of the rejection of claim 9 is earnestly solicited.

Claims 11 and 12 depend from claim 9 and add further distinguishing elements. Claims 11 and 12 are thus also submitted to be allowable. Withdrawal of the rejection of claims 11 and 12 is also earnestly solicited.

Claim Rejections - 35 U.S.C. § 103:

Claims 1, 2, and 5-8 were rejected under 35 U.S.C. § 103 as being unpatentable over Park in view of Khan et al., US 6,088,624 (hereinafter "Khan"). The rejection is traversed. Reconsideration is earnestly solicited.

Claim 1 recites:

A storage device or medium storing input/output units each including program block data obtained by dividing the machining program so that divided portions of the machining program are stored in respective ones of the input/output units.

Park neither teaches, discloses, nor suggests a storage device or medium storing input/output units each including program block data obtained by *dividing* the machining program so that divided portions of the machining program are stored in respective ones of the input/output units, as discussed above with respect to the rejection of claim 9. Khan does not either, and thus cannot make up for the deficiencies of Park with respect to claim 1. Thus, even if Park and Khan were combined as proposed in the final Office Action, the claimed invention would not result. Claim 1 is thus submitted to be allowable. Withdrawal of the rejection of claim 1 is earnestly solicited.

Claims 2 and 5-8 depend from claim 1 and add further distinguishing elements. Claims 2 and 5-8 are thus also submitted to be allowable. Withdrawal of the rejection of claims 2 and 5-8 is also earnestly solicited.

Claims 3 and 4:

Claims 3 and 4 were rejected under 35 U.S.C. § 103 as being unpatentable over Park in view of Khan, and further in view of Yamauchi et al., US 5,258,905 (hereinafter "Yamauchi"). The rejection is traversed. Reconsideration is earnestly solicited.

Claims 3 and 4 depend from claim 1 and add further distinguishing elements. Neither Park nor Khan teach, disclose, or suggest "a storage device or medium storing input/output units each including program block data obtained by dividing the machining program so that divided portions of the machining program are stored in respective ones of the input/output units," as discussed above with respect to the rejection of claim 1. Yamauchi does not either, and thus cannot make up for the deficiencies of Park with respect to claims 3 and 4. Thus, even if Park, Khan and Yamauchi were combined as proposed in the final Office Action, the claimed invention would not result. Claims 3 and 4 are thus submitted to be allowable. Withdrawal of the rejection of claims 3 and 4 is earnestly solicited.

Claim 10:

Claim 10 was rejected under 35 U.S.C. § 103 as being unpatentable over Park in view of Yamauchi. The rejection is traversed. Reconsideration is earnestly solicited.

Claim 10 depends from claim 9 and adds further distinguishing elements. Neither Park nor Yamauchi teach, disclose, or suggest a storage device storing a plurality of input/output units, each of the input/output units storing program data obtained by *dividing* the machining program so that divided portions of the machining program are stored in respective ones of the input/output units, as discussed above with respect to the rejection of claims 3 and 4. Thus, even if Park and Yamauchi were combined as proposed in the final Office Action, the claimed invention would not result. Claim 10 is thus submitted to be allowable. Withdrawal of the rejection of claims 10 is earnestly solicited.

Conclusion:

Accordingly, in view of the reasons given above, it is submitted that all of claims 1-13 are allowable over the cited references. Allowance of all claims 1-13 and of this entire application is therefore respectfully requested.

If there are any formal matters remaining after this response, the Examiner is requested to telephone the undersigned to attend to these matters.

Serial No. 10/613,011

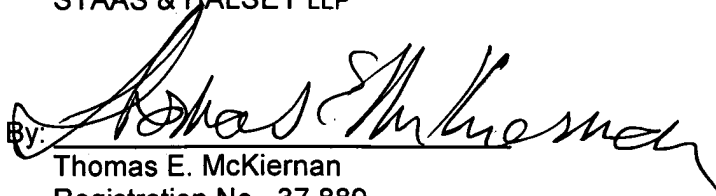
If there are any additional fees associated with filing of this Amendment, please charge the same to our Deposit Account No. 19-3935.

Respectfully submitted,

STAAS & HALSEY LLP

Date: 08 FEB 06

By:



Thomas E. McKiernan
Registration No. 37,889

1201 New York Ave, N.W., Suite 700
Washington, D.C. 20005
Telephone: (202) 434-1500
Facsimile: (202) 434-1501